DiffusionFF: A Diffusion-based Framework for Joint Face Forgery Detection and Fine-Grained Artifact Localization (Supplementary Material)

Siran Peng[1,2*] Haoyuan Zhang[2,1*] Li Gao[3] Tianshuo Zhang[2,1]
Xiangyu Zhu[1,2] Bao Li[1,2] Weisong Zhao[4] Zhen Lei[1,2,5,6†]
[1]MAIS, Institute of Automation, Chinese Academy of Sciences
[2]School of Artificial Intelligence, University of Chinese Academy of Sciences
[3]China Mobile Financial Technology Co., Ltd. [4]IIE, Chinese Academy of Sciences
[5]CAIR, HKISI, Chinese Academy of Sciences [6]SCSE, FIE, M.U.S.T
{pengsiran2023, zhanghaoyuan2023, zhen.lei}@ia.ac.cn

Abstract

In this supplementary material, we provide additional implementation details, extended experimental results, and further discussions to complement our main paper.

1. Additional Implementation Details

1.1. SBI

The SBI framework [17] synthesizes pseudo-fake facial images from real samples by simulating the deepfake generation process. It consists of two main components: the Source-Target Generator (STG) and the Mask Generator (MG). The STG applies a series of image transformations to create a pseudo-source and a pseudo-target image, while the MG produces a blending mask based on pre-detected facial landmarks, with augmentation to enhance diversity. The pseudo-source and pseudo-target images are then combined using the blending mask to generate a pseudo-fake facial image. In this study, we employ the SBI framework to augment the training data by integrating SBI-generated samples with those from the original FF++ dataset [16]. A key advantage of this approach is that the pseudo-fake images are perfectly aligned with their real counterparts, allowing for the precise computation of GT DSSIM maps.

1.2. Preprocessing

During training, we extract 32 frames from each real video and 8 frames from each fake video to maintain a balanced ratio between positive and negative samples, as each real video corresponds to four manipulated ones. Additionally, we sample 8 frames from each real video to generate pseudo-fake samples using the SBI framework, which are incorporated into the training set to enhance data diversity. During testing, we uniformly sample 32 frames per video. Facial landmarks are extracted using Dlib's 81-point predictor [10], which is utilized only during the training phase. For face detection, RetinaFace [4] is employed to obtain facial bounding boxes. During training, each detected face is cropped with a random margin ranging from 4% to 20%, while a fixed margin of 12.5% is applied during inference.

1.3. Data Augmentation

The image processing toolbox introduced in [1] is employed for data augmentation. Within the STG module of the SBI framework, transformations including RGBShift, HueSaturationValue, RandomBrightnessContrast, Downscale, and Sharpen are applied to generate pseudo source and target images. During training, all samples are augmented using operations such as ImageCompression, RGBShift, HueSaturationValue, and RandomBrightnessContrast. These augmentations expose the model to a broader range of visual variations, thereby enhancing its generalization capability.

1.4. Additional Details

The ConvNeXt-B detector [12] is trained on the FF++ dataset with SBI-based data augmentation. Training is conducted for 200 epochs using the AdamW optimizer [13], with a batch size of 64 and an initial learning rate of 5×10^{-5}. To promote stable convergence, a linear learning rate decay is applied starting from epoch 100. For the diffusion model, we employ a standard seven-stage U-Net architecture [15] with channel dimensions configured as [64, 64, 128, 256, 128, 64, 64]. For the noise schedule, the per-step variances $\{\beta_t\}_{t=1}^{T}$ increase linearly from 0.02 to 0.4. The GT DSSIM maps are generated using a local square window of size 7×7. For frames containing multiple detected

*Equal contribution.
†Corresponding author.

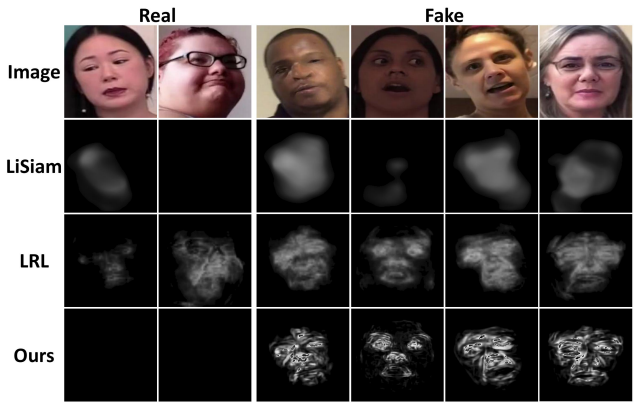

Figure 1. Qualitative DSSIM map estimation results on the DFDC dataset. Our DiffusionFF framework generates more fine-grained and detailed artifact localization maps than competing approaches. Note that GT DSSIM maps are unavailable for this dataset.

Table 1. Face forgery detection and DSSIM map estimation performance on the DeeperForensics-1.0 dataset. Our method consistently outperforms existing DSSIM-based approaches.

Method	Evaluation Metrics				
	PSNR↑	SSIM↑	LPIPS↓	FID↓	AUC ↑
LiSiam [19]	26.908	0.428	0.380	261.722	0.872
LRL [2]	27.941	0.507	0.338	244.558	0.925
DiffusionFF	**44.507**	**0.571**	**0.321**	**116.764**	**0.934**

faces, the classification model is applied to each face, and the maximum score is taken as the frame-level confidence.

2. Additional Experiments

2.1. Additional Qualitative Evaluation

In the DFDC [6] and DFDCP [5] datasets, the real and fake video pairs are not spatially aligned, which prevents the generation of GT DSSIM maps for quantitative evaluation. To assess the generalization capability of our DiffusionFF framework under this situation, we provide a qualitative comparison against existing DSSIM-based approaches on the DFDC dataset, as shown in Figure 1. Although GT DSSIM maps are unavailable, the visual results clearly indicate that DiffusionFF produces the most fine-grained and detailed outcomes, underscoring its superior effectiveness in generating high-quality artifact localization maps.

2.2. Results on DeeperForensics-1.0

We also evaluate our method on DeeperForensics-1.0 [9], a challenging dataset built upon the source videos of FF++. As shown in Table 1, DiffusionFF achieves superior quantitative results in both detection and localization. Notably, all methods are trained solely on the original FF++ dataset.

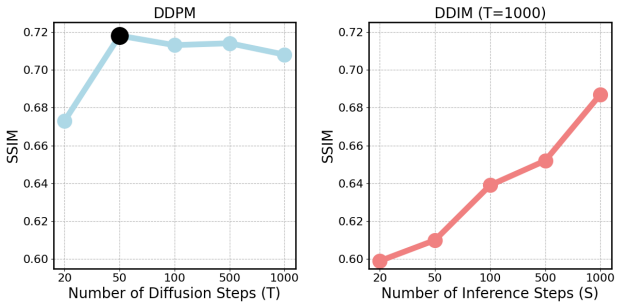

Figure 2. Ablation study on the training and inference settings of the diffusion model. The DDPM framework with $T = 50$ delivers the best performance, marked by the black dot in the left figure.

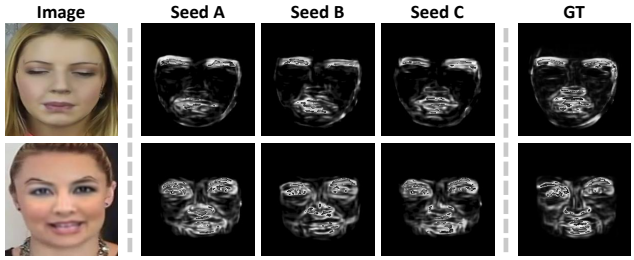

Figure 3. Ablation study on denoising seeds. Our method consistently generates stable localization maps across different seeds.

2.3. Additional Ablation Studies

2.3.1. Training & Inference of the Diffusion Model

In this study, we employ the standard DDPM framework [8], where the number of inference steps is set equal to the number of training diffusion steps (T). We begin by examining how varying T affects performance. The quantitative results shown on the left of Figure 2 indicate that our model achieves optimal performance at $T = 50$. This outcome contrasts with the common expectation that larger T values generally lead to better generation quality. We hypothesize that this discrepancy arises because our conditional DSSIM estimation task differs fundamentally from traditional RGB image generation, thereby shifting the optimal hyper-parameter landscape. To further validate our framework choice, we also evaluate the Denoising Diffusion Implicit Model (DDIM) framework [18], which enables faster inference by skipping diffusion steps. In this experiment, we use a model trained with $T = 1000$ and perform inference using S sampling steps. As shown on the right of Figure 2, while DDIM performance improves with increasing S, it consistently underperforms the DDPM $T = 1000$ baseline, let alone the superior DDPM $T = 50$ setup. Collectively, these findings firmly validate our choice to use the

Method	Test Set AUC (%) ↑			
	CDF2	DFDC	DFDCP	FFIW
Single-Stage	95.47	83.54	90.27	87.69
Two-Stage	**97.24**	**85.05**	**92.56**	**88.56**

Table 2. Ablation study on the two-stage training strategy.

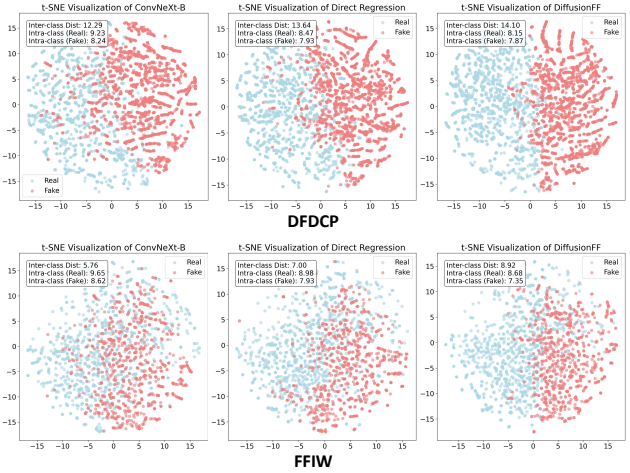

Figure 4. t-SNE visualizations of feature representations on the DFDCP and FFIW datasets. Compared to ConvNeXt and direct regression, DiffusionFF yields more distinct and well-separated clusters, demonstrating its superior discriminative capability.

DDPM framework with a small timestep of $T = 50$.

2.3.2. Denoising Seeds

We evaluate the impact of random seed variation on DSSIM map estimation performance. As illustrated in Figure 3, our DiffusionFF framework consistently generates stable and reliable artifact localization maps across different seeds.

2.3.3. Two-Stage Training Strategy

We compare our two-stage training strategy against a single-stage baseline that jointly optimizes DSSIM map estimation and binary classification. As shown in Table 2, our approach yields significantly better results, demonstrating the effectiveness of training these two tasks separately.

2.3.4. Additional t-SNE Visualizations

As illustrated in Figure 4, additional t-SNE visualizations on the DFDCP and FFIW datasets show that DiffusionFF achieves a clearer separation between real and fake samples than the ConvNeXt backbone and the direct regression strategy, demonstrating the effectiveness of our method.

3. Discussions

In this section, we first discuss the requirement of aligned pairs and the computational cost associated with our method. Subsequently, we discuss three critical questions: (1) Can GT DSSIM maps serve as inputs to the artifact feature extractor during our second training stage? (2) Is the ControlNet framework [20] capable of estimating DSSIM maps? and (3) What causes the joint training of the forgery detector and diffusion model from scratch to fail? Finally, we summarize the strengths and limitations of our work.

3.1. Requirement of Aligned Pairs

While generating GT DSSIM maps requires aligned real-fake image pairs, this condition applies solely during training; inference requires only a single target image. Importantly, this requirement does not limit the applicability of our method. In the broader domain of image forgery detection, datasets like CoCoGlide [7] and SAN [3] naturally provide such aligned pairs. For datasets lacking them, we can synthesize aligned fake images from real ones using methods like SBI [17]. This flexible adaptation guarantees the creation of GT DSSIM maps across diverse scenarios.

3.2. Computational Cost

During inference, the isolated forgery detector, the denoising diffusion model, and the complete DiffusionFF pipeline require 15, 1972, and 1990 GFLOPs, with corresponding execution times of 0.01s, 0.33s, and 0.34s per frame. While we acknowledge the inherent computational overhead of diffusion models, our modular architecture offers significant deployment flexibility. The decoupled detector can operate independently for high-throughput, real-time filtering (0.01s). Conversely, in scenarios where explainability is paramount, the full pipeline can be activated. Although this increases the computational cost (0.34s), it provides unique fine-grained artifact localization and SOTA detection performance, presenting a highly justifiable trade-off.

3.3. GT DSSIM Maps As Inputs During Training

In Figure 2 of the main text, we demonstrate a clear positive correlation: the higher the quality of the DSSIM maps integrated into the detection network, the greater the resulting performance gains. This observation naturally leads to the question: what if GT DSSIM maps were used as inputs to the artifact feature extractor during our second training stage? To explore this, we conduct an experiment using GT DSSIM maps and obtain a 96.72% AUC on the CDF2 dataset [11], which is lower than the 97.24% AUC achieved with diffusion-estimated maps. This performance drop is primarily due to a mismatch in value distributions: GT DSSIM maps (e.g., all zeros for real samples) differ significantly from those estimated by the diffusion model. As a result, a model trained with GT DSSIM maps struggles to generalize when faced with diffusion-estimated inputs during inference, ultimately leading to degraded performance.

3.4. ControlNet for DSSIM Map Estimation

ControlNet is a widely adopted framework for conditional image generation that integrates an auxiliary conditioning network to guide a pretrained, frozen Stable Diffusion model [14]. **In our study, we attempted to train a ControlNet to estimate DSSIM maps but encountered training collapse, where the network produces identical, content-irrelevant outputs regardless of input variation**. We attribute this failure to a fundamental domain mismatch: the pretrained Stable Diffusion model, optimized for natural RGB image synthesis, struggles to adapt to single-channel, grayscale DSSIM maps due to its strong prior knowledge. To overcome this limitation, we propose an inverted strategy that contrasts with the conventional ControlNet paradigm. Instead of training a new conditioning network, we employ a pretrained forgery detector as a fixed conditioning model, leveraging its capacity to capture forgery-related features. By freezing the conditioning network and training the diffusion model instead, we shift the learning focus to the generative model itself. This reversal allows the diffusion model to effectively utilize the detector's specialized features, enabling precise and stable estimation of DSSIM maps.

3.5. Training Detector and Diffusion From Scratch

As noted in Section 3.2.2, jointly training the forgery detector and diffusion model from scratch leads to training collapse. Here, we analyze this failure in detail. **During training, the model converges to a trivial solution: producing entirely black images regardless of input.** This collapse arises because the GT DSSIM maps for real images are uniformly black. Consequently, predicting black results in zero loss for real samples, allowing the model to minimize training loss without learning the more complex task of detecting and localizing artifacts in fake images. This behavior highlights that the DSSIM estimation task intrinsically depends on forgery-related features. Without explicit guidance from these features, the diffusion model fails to learn the distinction between real and fake faces. To address this issue, we employ a pretrained forgery detector to extract forgery-specific features that guide the diffusion model in generating fine-grained artifact localization maps.

3.6. Strengths & Limitations

3.6.1. Strengths

The proposed DiffusionFF framework exhibits three key advantages. First, it enables precise localization of fine-grained forgery clues, thereby enhancing model explainability and fostering greater user trust. Second, by integrating the generated artifact localization maps into the detection pipeline, DiffusionFF significantly improves overall detection performance. Third, owing to its strong generality, DiffusionFF can serve as an explainable, plug-and-play enhancement for existing face forgery detection models.

3.6.2. Limitations

Despite its many advantages, the DiffusionFF framework also presents two limitations. First, the inference process of the diffusion model is time-consuming. Second, our method is not designed for entirely synthesized images.

References

[1] Alexander Buslaev, Vladimir I Iglovikov, Eugene Khvedchenya, Alex Parinov, Mikhail Druzhinin, and Alexandr A Kalinin. Albumentations: fast and flexible image augmentations. *Information*, 11(2):125, 2020. 1

[2] Shen Chen, Taiping Yao, Yang Chen, Shouhong Ding, Jilin Li, and Rongrong Ji. Local relation learning for face forgery detection. In *Proceedings of the AAAI conference on artificial intelligence (AAAI)*, pages 1081–1088, 2021. 2

[3] Tao Dai, Jianrui Cai, Yongbing Zhang, Shu-Tao Xia, and Lei Zhang. Second-order attention network for single image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 3

[4] Jiankang Deng, Jia Guo, Evangelos Ververas, Irene Kotsia, and Stefanos Zafeiriou. Retinaface: Single-shot multi-level face localisation in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1

[5] Brian Dolhansky, Russ Howes, Ben Pflaum, Nicole Baram, and Cristian Canton Ferrer. The deepfake detection challenge (dfdc) preview dataset. *arXiv preprint arXiv:1910.08854*, 2019. 2

[6] Brian Dolhansky, Joanna Bitton, Ben Pflaum, Jikuo Lu, Russ Howes, Menglin Wang, and Cristian Canton Ferrer. The deepfake detection challenge (dfdc) dataset. *arXiv preprint arXiv:2006.07397*, 2020. 2

[7] Fabrizio Guillaro, Davide Cozzolino, Avneesh Sud, Nicholas Dufour, and Luisa Verdoliva. Trufor: Leveraging all-round clues for trustworthy image forgery detection and localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20606–20615, 2023. 3

[8] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems (NeurIPS)*, 33:6840–6851, 2020. 2

[9] Liming Jiang, Ren Li, Wayne Wu, Chen Qian, and Chen Change Loy. Deeperforensics-1.0: A large-scale dataset for real-world face forgery detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[10] Davis E King. Dlib-ml: A machine learning toolkit. *The Journal of Machine Learning Research*, 10:1755–1758, 2009. 1

[11] Yuezun Li, Xin Yang, Pu Sun, Honggang Qi, and Siwei Lyu. Celeb-df: A large-scale challenging dataset for deepfake forensics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 3

[12] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11976–11986, 2022. 1

[13] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 1

[14] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 10684–10695, 2022. 4

[15] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*, pages 234–241. Springer, 2015. 1

[16] Andreas Rossler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Niessner. Faceforensics++: Learning to detect manipulated facial images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 1

[17] Kaede Shiohara and Toshihiko Yamasaki. Detecting deepfakes with self-blended images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18720–18729, 2022. 1, 3

[18] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 2

[19] Jian Wang, Yunlian Sun, and Jinhui Tang. Lisiam: Localization invariance siamese network for deepfake detection. *IEEE Transactions on Information Forensics and Security*, 17:2425–2436, 2022. 2

[20] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF international conference on computer vision (CVPR)*, pages 3836–3847, 2023. 3